

REMARKS

Claims 1-20, 22 and 24 were pending. No claims have been amended, added or canceled. Accordingly, claims 1-20, 22 and 24 remain pending subsequent entry of the present amendment.

In the present Office Action claims 1-20, 22 and 24 stand rejected under 35 U.S.C. § 103(a) as being anticipated by previously cited reference Matic, et al. ("Predictive Payout Delay Adaptation for Voice over Internet") (hereinafter "Matic") in view of U.S. Patent No. 6,622,173 (hereinafter "Luke"). Applicant has carefully reviewed the references, and considered the examiner's comments. However, Applicant submits each of the pending claims recite features neither disclosed nor suggested by the cited art. Accordingly, Applicant traverses the above rejections and requests reconsideration.

In the previous Office Action, it was suggested that the cited reference Luke teaches the features of the Packet ID as recited in claim 1 by disclosing a PktNum. Here, in the present Final Office Action, an alternative explanation is provided. In particular, it is now suggested that the "Packet" disclosed by Luke, and not the PktNum as previously suggested, teaches the features as claimed. In the present Office Action, the Examiner states:

"The first point of contention involves the claimed Packet ID. The applicant claims that neither prior arts teach the claimed Packet ID and explained further how Packet ID is different from the PktNum taught by Luke. A further review of the prior art has made clear though that Luke teaches the claimed Packet ID and it is deemed equivalent to the numbers in figure 3, box 22, left column under header "Packet"." (emphasis added).

However, Applicant disagrees. The "Packet" of Luke, which is an identifier (ID) of a received packet, is not equivalent to the claimed Packet ID which is an ID of a future

packet. Claim 1 recites a packet buffering system for predictively processing data packets that comprises:

- “at least one input port for receiving data packets from a plurality of sources, wherein the received data packets arrive from the plurality of data flows, interspersed;
- at least one output port for sending out data packets to a plurality of destinations;
- a packet predictor, coupled to said at least one input port, for predicting information about a future packet in any one of the plurality of data flows based on history of previously received packets from the plurality of data flows, said history stored in a memory coupled to said packet predictor;
- a plurality of queues for storing packets received from said plurality of sources, and for storing said predicted information about said future packet;
- direction logic, coupled to said packet predictor, for generating a Packet ID for said future packet, wherein said Packet ID is stored in one of said plurality of queues;
- buffer logic, coupled to said packet predictor, for accessing said memory and for validating said predicted information about said future packet based on said access to said memory; and
- a processing core, coupled to said plurality of queues, wherein if said buffer logic validates said predicted information, notification is made to said direction logic which passes said Packet ID for said future packet to said processing core to initiate speculative processing.” (emphasis added).

As can be seen above, claim 1 recites a Packet ID generated for a future packet – in other words, the recited Packet ID is generated for a packet not yet received. In contrast, the Packet of Luke is an ID of an already received packet. Applicant submits these are clearly not equivalent and the claim is distinguishable for at least these reasons.

In addition to the above, claim 1 also recites “a packet predictor, coupled to said at least one input port, for predicting information about a future packet in any one of the plurality of data flows based on history of previously received packets from the plurality

of data flows, said history stored in a memory coupled to said packet predictor.”
Regarding prediction related information, Luke discloses the following:

“The receiving proxy 22 comprises a packet receiver buffer acting as a lookup table which associates previously seen packets with prediction checksums and contains a reference to the contents of the previously received message which is held locally in message store 24 . PktNum represents the number of packets in the prediction associated with the checksum. For example, if the receiving proxy 22 receives a packet 1000, it predicts the occurrence of Message 2 and transmits a checksum, 101, which represents the following 3 packets which are stored locally and accessible to the receiving proxy 22 .

It should be seen that the construction of the packet receiver buffer is best tuned to the operating conditions of the messaging system 10, 20. In the present example, where messages are assumed to comprise approximately 5 packets, a three bit PktNum is sufficiently large to cope with the number of packets the system is to make a prediction about. The system also needs to decide about the packet within a message it is to make a prediction. In the present example, the second packet of a message is chosen as the one on which to base a prediction and the checksum is used to predict the remaining packets.” (Luke, col. 2 lines 38-59). (emphasis added).

As can be seen above, the “packet” in receiving proxy 22 corresponds to previously seen packets and not future packets. This is further supported by FIGS. 5-8 and in the description associated with these figures:

“FIG. 5 shows the transmission of the first packet with its location in the transmission buffer being de-allocated. FIG. 6 shows a failure to identify a match within the packet receiver buffer for the first packet. As there is no match, the receiver proxy 22 simply either explicitly requests or waits for the next packet.

FIG. 7 shows the transmission of the second packet and FIG. 8 shows the response of the receiver proxy 22 to the receipt of the second packet. Having identified a match in the packet receiver buffer, the proxy 22 predicts an occurrence of Message 2 and sends the checksum together with the number of packets covered by the checksum. In FIG. 9, the transmitter proxy 12 verifies the checksum and transmits a prediction correct signal to the receiver proxy. The proxy 12 will not send further packets for Message 2 and as such is free to flush the

packet transmission buffer of the unsent packets. On the other hand, the receiver proxy 22 is able to reconstruct the original message using a stored copy of at least the remainder of Message 2.” (Luke, col. 3 lines 23-41).

As described earlier in Luke, prediction begins with the second packet of a message, which is sent in FIG. 7 with an identifier 1000. This identifier existed in transmitting proxy 12, as shown in FIG. 5-6, prior to being sent. After being sent, this same identifier 1000 is compared to identifiers of previously seen packets in receiving proxy 20 as shown in FIG. 7. There is no logic “for generating a Packet ID for said future packet” as recited in the claim. The identifier 1000 in Luke was previously seen by receiver proxy 22, and subsequently stored by receiver proxy 22. Then identifier 1000 is seen again by proxy 22. A checksum and PktNum are calculated by proxy 22 and sent to transmitting proxy 12. Identifier 1000 remains stored in proxy 22 for possible future comparisons. Accordingly, the value of identifier 1000 is not equivalent to the recited features.

Also, the direction logic of claim 1 “passes said Packet ID for said future packet to said processing core to initiate speculative processing”. In contrast, the identifier 1000 in Luke is not passed to initiate speculative processing. Rather, in Luke a calculated checksum and PktNum are passed from the receiving proxy 20 to the transmitting proxy 12 in order to initiate speculative processing as described above and shown in FIG. 8. FIG. 8 does not show any passing of the identifier 1000. If the transmission proxy 12 determines the prediction is correct, then the receiving proxy 20 completes the message from the messages previously stored in the Message Store shown in FIG. 3. Therefore, the “packet” disclosed by Luke does not teach a packet ID as recited in claim 1. For at least each of the above reasons, claim 1 is patently distinct from the cited reference Luke.

Still further, claim 1 recites logic within a packet buffering system such as:

“... a packet predictor, coupled to said at least one input port, for predicting information about a future packet in any one of the plurality of data flows based on history of previously received

packets from the plurality of data flows, said history stored in a memory coupled to said packet predictor;
... buffer logic, coupled to said packet predictor, for accessing said memory and for validating said predicted information about said future packet based on said access to said memory”

Claim 1 recites validating predicted information by first accessing a memory that stores a history of previously received packets. The step of accessing the memory occurs prior to validating the predicted information, since validating is based on “said access to said memory”. The memory recited in claim 1 is specific and it is not a simply a “memory of some form”. The memory that is being accessed for validating predicted information has antecedent basis in the claim and is recited as storing a history of previously received packets.

In the present Office Action, it is suggested that Luke discloses these features of claim 1, since Luke discloses the access of “memory of some form” prior to validating predicted information. For example, the Examiner states:

“The second point of contention involves the claimed validating step. The applicant argues that Luke teaches accessing the Message Store after the checksum is validated (wherein the checksum has been deemed equivalent to the validation step) whereas the claim accesses memory first and then validates. This would be correct, only if the Luke design didn't access at least one memory before validating (using checksum). However, it is impossible for a digital design to read, write, validate, edit, or handle data without it being in memory of some form. Hence, the claimed accessing of memory prior to validating is inherent.” (emphasis added).

The “memory of some form” that is disclosed by Luke to be accessed prior to validating predicted information, such as the checksum, is the transmitting proxy 12. This is shown, for example, in FIG. 9 of Luke. However, transmitting proxy 12 stores packets that are broken down from messages and are to be sent to a receiving destination. Transmitting proxy 12 does not store a “history of previously received packets”. Rather, the Message Store 24 [shown in FIG. 3 without the reference numeral which appears in

the description] in receiving proxy 20 stores a “history of previously received packets”. Below, Luke discloses that transmitting proxy 12 is used to validate predicted information (i.e. checksum) and stores packets to be sent - not a history of previously seen packets. Message store 24 stores a history of previously seen packets, and is accessed after validation of the checksum as described in the following:

“In FIG. 9, the transmitter proxy 12 verifies the checksum and transmits a prediction correct signal to the receiver proxy.” (Luke, col. 3 lines 34-36).

“Referring now to FIG. 1, which shows a conventional messaging system where messages are broken down into packets and sent by a transmitting system 10 . . .

FIG. 2 shows a proxy system according to a preferred embodiment of the invention complementing such an existing message system 10, 20. In this case, a transmitting proxy 12, comprising a packet transmission buffer, is associated with the transmitting system 10 . . .” (Luke, col. 2 lines 12-22).

“The receiving proxy 22 . . . contains a reference to the contents of the previously received message which is held locally in message store 24.” (Luke, col. 2 lines 38-42).

“. . . a prediction correct signal to the receiver proxy. The proxy 12 will not send further packets for Message 2 and as such is free to flush the packet transmission buffer of the unsent packets. On the other hand, the receiver proxy 22 is able to reconstruct the original message using a stored copy of at least the remainder of Message 2.” (Luke, col. 3 lines 36-41)

As can be seen, Luke discloses an access of a “memory of some form” (i.e. transmitting proxy 12) prior to validation of the checksum. However, this is clearly not the memory as recited in the claim which stores packet history information. Luke also discloses an access of a memory (i.e. memory store 24) that stores a history of previously seen packets occurs after validation. However, this memory is accessed subsequent to validation and does not disclose the features as recited in the claim. Therefore, Luke does not disclose the features of the claimed invention as recited in claim 1. For at least these additional reasons, claim 1 is patently distinguishable from the cited reference.

In view of the above discussion, Applicant submits the above features are neither disclosed nor suggested by Luke. Neither are the above features disclosed or suggested by Matic. Therefore, the above features of claim 1 are nowhere present in either Luke or Matic taken alone or in combination, and claim 1 is believed to be patentably distinct from the cited references. Claim 17 is distinguishable for similar reasons. As each of the dependent claims include the features of the independent claims on which they depend, each of the dependent claims are patentably distinct for at least the above reasons.

In addition to the above, the dependent claims recite still further features not disclosed or suggested by the cited art, taken either singly or in combination. For example, claim 6 recites the additional features:

“wherein said packet predictor predicts specific characteristics, comprising one or more of packet type, packet flow identification, sender information, destination information, and packet size for said future packet.”

In the present Office Action, it is suggested that Matic discloses these features. In particular, the Examiner states:

“The third point of contention involves the claimed traits of claims 6 and 15. The claim cites the packet predictor as predicting specific characteristics comprising one or more of packet type, packet flow identification, sender information, destination information, and packet size for said future packet. The applicant contends that neither prior art teach such features, the examiner disagrees. Section 1, p. 348, 2nd column, 6th paragraph of the Matic art teaches, “To achieve this goal we need to know characteristics of jitter in advance. Hence, a main component of smoother on a receiver side is a jitter predictor.” This means jitter is a receiving characteristic/information making it equivalent to the claimed destination information.”

Applicant does not agree that predicting jitter is equivalent to predicting destination information of a packet. Within the context of the present art, those skilled understand the

meaning of such terms as packet sender (source) information and destination information. As described in the present application (para. 33 of the corresponding publication 20030041168):

“When a packet arrives, predictor 202 generates a prediction of the nature of the next packet to arrive based on data in MEM 204. Information predicted can include several separable components of a next packet, all of which, or some of which may be predicted depending on desired configuration. For example, header information includes typically some or all of source information, destination information, packet flow information, data protocol information, packet size information, and media type information.”

In contrast to the above, Matic discusses a jitter predictor. However, jitter concerns varying packet inter-arrival time and in no way concerns destination information of a packet. Accordingly, Applicant submits the combination of references do not disclose the additional features of claim 6, and claim 6 is patentable over the cited art. Claim 15 is patentable for similar reasons as well.

Applicant believes all claims to be patentably distinguished from the cited art for at least the above reasons.

CONCLUSION

Applicant submits the application is in condition for allowance, and an early notice to that effect is requested.

Respectfully submitted,

/James W. Huffman/

James W. Huffman

Reg. No. 35,549

ATTORNEY FOR APPLICANT(S)

Date: 3/10/08